

CLAIMS

1. A system for handling a completion process in a remote data memory access (RDMA) environment having a RequestOut channel and a ResponseOut channel, comprising:

a descriptor list for each channel, wherein each descriptor list includes a message descriptor for each message in the channel;

an update mechanism for updating a message length field in the message descriptor with a sequence number of a last byte in the message whenever a channel swap occurs between the RequestOut channel and the ResponseOut channel;

an acknowledgement (Ack) completion system that examines values in a completion context and compares a sequence number of a next to complete message with a last acknowledged sequence number to determine if the message should be completed; and

a read request completion system that performs completion of a read request.

2. The system of claim 1, wherein the acknowledgement completion system includes a series of repeating logic steps that are separately applied to each of the RequestOut channel and the ResponseOut channel.

3. The system of claim 2, wherein the repeating logic steps include:

concluding that the processing of the completion process is finished if the sequence number of the next to complete message is invalid;

concluding that the processing of the completion process is finished if the sequence number of the next to complete message is greater than the last acknowledged sequence number; and

completing the message in the channel if the sequence number of the next to complete message is less than or equal to the last acknowledged sequence number.

4. The system of claim 3, wherein the repeating logic steps include the further step of: terminating the completion process in the RequestOut channel if the RequestOut channel is waiting for a completion of a read request.

5. The system of claim 3, wherein the step of completing the message in the channel further includes the steps of:

updating the sequence number of the last completed message with the sequence number of the next to complete message; and

updating the sequence number of the next to complete message with a last sequence number of a next message in the channel.

6. The system of claim 5, wherein the step of completing the message in the channel further includes the steps of:

performing completion of the operation if the completed message is not a read request message;

if the message is a read request message, waiting for reception and delivery of a read response to perform completion before performing completion, and then setting a pending read request bit in the completion context.

7. The system of claim 2, wherein the read request completion system provides a second series of logic steps that include:

completing any requests preceding the read request;

completing the read request; and

completing any requests following the read request.

8. The system of claim 7, wherein the last two steps of the second set of logic steps are repeated N times, wherein N is a value stored in the completion context that represents the number of completed read requests.

9. The system of claim 1, further comprising a system for handling a retransmit request that includes a third series of logic steps for locating a segment to retransmit, wherein the steps include:

performing a completion operation to ensure that there is no pending completion;

identifying a candidate message for both the RequestOut channel and the ResponseOut channel;

selecting a message carrying the segment to transmit from the two candidate messages; and

determining a location of a pointer descriptor that refers to a beginning of the segment to retransmit.

10. The system of claim 9,

wherein if the RequestOut channel is waiting for completion of a pending read request, the candidate message in the RequestOut channel comprises either a first not completed message in the RequestOut channel;

wherein if the RequestOut channel is not waiting for completion of a pending read request, the candidate message in the RequestOut channel is given by a next-to-complete (N2C) pointer; and

wherein the candidate message in the ResponseOut channel is given by a next-to-complete (N2C) pointer.

11. The system of claim 9, wherein the message carrying the segment to transmit is the candidate message that resides in the channel having the lowest sequence number for the next to complete message.

12. The system of claim 9, wherein the location of the pointer descriptor that refers to the beginning of the segment to retransmit is one more than the maximum of:

the sequence number of the last completed message in the RequestOut channel;

and

the sequence number of the last completed message in the ResponseOut channel.

13. A method for handling a completion process in a remote data memory access (RDMA) environment having a RequestOut channel and a ResponseOut channel, including performing an acknowledgement completion on each channel with the steps of:

concluding that the processing of the completion process is finished if the sequence number of a next to complete message is invalid;

concluding that the processing of the completion process is finished if the sequence number of the next to complete message is greater than the last acknowledged sequence number;

completing the message in the channel if the sequence number of the next to complete message is less than or equal to the last acknowledged sequence number; and

terminating the completion process in the RequestOut channel if the RequestOut channel is waiting for a completion of a read request.

14. The method of claim 13, wherein the step of completing the message in the channel includes the further steps of:

updating the sequence number of a last completed message with the sequence number of the next to complete message; and

updating the sequence number of the next to complete message with a last sequence number of a next message in the channel.

15. The method of claim 14, wherein the step of completing the message in the channel includes the further steps of:

if the completed message is not a read request message, performing completion of the operation; and

if the message is a read request message, waiting for a read response to perform completion and setting a pending read request bit.

16. The method of claim 13, further comprising a read request completion method that includes the steps of:

completing any requests preceding the read request;

completing the read request;

completing any requests following the read request; and

repeating the previous two steps N times, wherein N is a value that represents the number of completed read requests.

17. A method for locating a segment to retransmit for a retransmit request in a remote data memory access (RDMA) environment having a RequestOut channel and a

ResponseOut channel, comprising:

- performing a completion operation;

- identifying a candidate message for both the RequestOut channel and the ResponseOut channel;

- selecting a message carrying the segment to transmit from the two candidate messages; and

- determining a location of a pointer descriptor that refers to a beginning of the segment to retransmit.

18. The method of claim 17,

- wherein if the RequestOut channel is waiting for completion of a pending read request, the candidate message in the RequestOut channel comprises a first not completed message in the RequestOut channel;

- wherein if the RequestOut channel is not waiting for completion of a pending read request, the candidate message in the RequestOut channel is given by a next-to-complete (N2C) pointer; and

- wherein the candidate message in the ResponseOut channel is given by a next-to-complete (N2C) pointer.

19. The method of claim 18, wherein the message carrying the segment to transmit is the candidate message that resides in the channel having the lowest sequence number for the next to complete message.

20. The method of claim 18, wherein the location of the pointer descriptor that refers to the beginning of the segment to retransmit is one more than the maximum of:

the sequence number of the last completed message in the RequestOut channel;

and

the sequence number of the last completed message in the ResponseOut channel.

21. A system for handling a transmit process in a remote data memory access (RDMA) environment having a RequestOut channel and a ResponseOut channel, comprising:

- a descriptor list for each channel, wherein each descriptor list includes a message descriptor for each message in the channel; and
- an update mechanism for updating a message length field in the message descriptor with a sequence number of a last byte in the message whenever a channel swap occurs between the RequestOut channel and the ResponseOut channel.